

Claims:

This listing of Claims replaces all prior versions and listings of Claims in the application:

1. **(Currently Amended)** A method for communication between a first computer operating in a first object-oriented run-time environment and a second computer operating in a second, different object-oriented run-time environment, the method comprising:  
**receiving sending** a first message with an object identification and an action identification from the first computer to the second computer;  
identifying an object in the second run-time environment according to the object identification;  
verifying an existence of an action, according to the action identification, in the identified object in the second run-time environment;  
determining an action representation of an action, according to the action identification, in the second run-time environment for the identified object; and  
executing the action using the action representation, **wherein executing the action includes:**  
**converting a request identification that is part of the action identification to a second representation for the second run-time environment using a look-up table; and**  
**inserting the second representation into the second application.**
2. **(Canceled)**
3. **(Canceled)**
4. **(Original)** The method of claim 1 further comprising returning to the first computer a second message as a confirmation message with an object identification and a response identification.

5. **(Original)** The method of claim 4 further comprising displaying, using the first computer, at least a portion of the response identification.
6. **(Previously Presented)** The method of claim 1, wherein executing the action further comprises:
  - extracting a second property representation of a property identified by the action identification;
  - converting the second property representation to a first property representation for the first run-time environment; and
  - returning to the first computer a second message as a result message with an object identification and a response identification, the response identification indicating the first property representation for the first run-time environment.
7. **(Previously Presented)** The method of claim 6 further comprising displaying, using the first computer, at least a portion of the response identification.
8. **(Currently Amended)** The method of claim 1, **further comprising:**
  - receiving a second message with a second object identification and a second action identification from the first computer to the second computer;**
  - identifying a second object in the second run-time environment according to the second object identification;**
  - determining a second action representation of a second action, according to the second action identification, in the second run-time environment for the identified second object; and**
  - executing the second action using the second action representation,** wherein executing the **second** action further comprises:
    - converting a function identification and a parameter identification of the **second** action identification to function and parameter representations for the second run-time environment;

performing a function that is identified by the second action identification using the function and parameter representations for the second run-time environment;

converting parameters that result from performing the function into parameter representations for the first run-time environment; and

returning a third second message to the first computer with an object identification and a response identification, with the response identification indicating the parameter representations.

9. **(Original)** The method of claim 8 wherein converting parameters uses a look-up table.

10. **(Currently Amended)** A computer program product used in a communication system of a first computer with a first object-oriented run-time environment and a second computer with a second, different object-oriented run-time environment, wherein the first computer sends a first message with an object identification and an action identification to the second computer, the computer program product embodied on a carrier and having computer code instructions to cause a processor of the second computer to interpret the first message, the instructions comprising:

code for identifying an object in the second run-time environment according to the object identification;

code for verifying the existence of an action, according to the action identification, in the identified object in the second run-time environment; **and**

code for determining a representation of an action, according to the action identification, in the second run-time environment for the identified object; and

code for executing the action using the representation, wherein the code for executing includes:

code for converting a request identification that is part of the action identification to a further representation for the second run-time environment; and

code for inserting the further representation into the second application.

11. **(Canceled)**
12. **(Previously Presented)** The computer program product of claim 10 wherein the instructions further comprise code for returning a second message as a confirmation message to the first computer, the second message including an object identification and a response identification.
13. **(Canceled)**

14. **(Currently Amended)** The computer program product of claim 10 **[[13]]** wherein the code for converting uses a look-up table.
15. **(Currently Amended)** The computer program product of claim 10 **[[12]]** wherein the code for executing comprises:
  - code for extracting a second property representation of a property identified by the action identification;
  - code for converting the second property representation to a first property representation for the first run-time environment; and
  - code for returning to the first computer a second message as a result message with an object identification and a response identification, the response identification indicating the first property representation for the first run-time environment.
16. **(Currently Amended)** The computer program product of claim 10 **[[12]]**, wherein the code for executing comprises:
  - code for converting a function identification and a parameter identification of the action identification to function and parameter representations for the second run-time environment;
  - code for performing a function that is identified by the action identification using the function and parameter representations for the second run-time environment;
  - code for converting parameters that result from performing the function into parameter representations for the first run-time environment; and
  - code for returning a second message to the first computer with an object identification and a response identification, with the response identification indicating the parameter representations.

17. **(Currently Amended)** A computer communication system comprising a first computer operating in a first object-oriented run-time environment and a second computer operating in a second, different object-oriented run-time environment, wherein the first computer sends a first message with an object identification and an action identification to the second computer, the second computer comprising:

a first module to identify an object in the second run-time environment according to the object identification;

a second module to verify an existence of an action identified in the action identification in the identified object in the second run-time environment;

a third module to determine a representation of the action in the second run-time environment for the identified object; and

a fourth module to execute the action by using the representation and to return a second message as confirmation message to the first computer, the second message with object identification and response identification, **wherein the fourth module is adapted to (a) convert a request identification that is part of the action identification to a further representation for the second run-time environment using a look-up table, and (b) insert the further representation into the second application.**

18. **(Canceled)**

19. **(Currently Amended)** The computer communication system of claim 17 wherein the fourth module is **further** adapted to:

extract a second property representation of a property identified by the action identification;

convert the second property representation to a first property representation for the first run-time environment; and

return to the first computer a second message as a result message with an object identification and a response identification, the response identification indicating the first property representation for the first run-time environment.

20. **(Currently Amended)** The computer communication system of claim 17 wherein the fourth module is **further** adapted to:

convert a function identification and a parameter identification of the action identification to function and parameter representations for the second run-time environment;

perform a function that is identified by the action identification using the function and parameter representations for the second run-time environment;

convert parameters that result from performing the function into parameter representations for the first run-time environment; and

return a second message to the first computer with an object identification and a response identification, with the response identification indicating the parameter representations.

- 21. (New) A computer program product comprising computer readable instructions operable when executed to:**

**receive a first message with an object identification and an action identification from a first computer with a first object-oriented run-time environment;**

**identify an object in a second computer with a second, different object-oriented run-time environment, according to the object identification;**

**verify the existence of an action, according to the action identification, in the identified object in the second run-time environment;**

**determine a representation of an action, according to the action identification, in the second run-time environment for the identified object; and**

**execute the action using the representation, wherein the instructions operable to execute the action comprise instructions operable to:**

**convert a request identification that is part of the action identification to a further representation for the second run-time environment; and**

**insert the further representation into the second application.**

22. (New) A computer program product comprising computer readable instructions operable when executed to:

receive a first message with an object identification and an action identification from a first computer with a first object-oriented run-time environment;

identify an object in a second computer with a second, different object-oriented run-time environment, according to the object identification;

verify the existence of an action, according to the action identification, in the identified object in the second run-time environment;

determine a representation of an action, according to the action identification, in the second run-time environment for the identified object; and

execute the action using the representation, wherein the instructions operable to execute the action comprise instructions operable to:

extract a second property representation of a property identified by the action identification;

convert the second property representation to a first property representation for the first run-time environment; and

return to the first computer a second message as a result message with an object identification and a response identification, the response identification indicating the first property representation for the first run-time environment.

23. (New) A computer program product comprising computer readable instructions operable when executed to:

receive a first message with an object identification and an action identification from a first computer with a first object-oriented run-time environment;

identify an object in a second computer with a second, different object-oriented run-time environment, according to the object identification;



verify the existence of an action, according to the action identification, in the identified object in the second run-time environment;

determine a representation of an action, according to the action identification, in the second run-time environment for the identified object; and

execute the action using the representation, wherein the instructions operable to execute the action comprise instructions operable to:

convert a function identification and a parameter identification of the action identification to function and parameter representations for the second run-time environment;

perform a function that is identified by the action identification using the function and parameter representations for the second run-time environment;

convert parameters that result from performing the function into parameter representations for the first run-time environment; and

return a second message to the first computer with an object identification and a response identification, with the response identification indicating the parameter representations.

24. (New) The method of claim 8, the second object identified in the second message being the first object identified in the first message.